

WHAT IS CLAIMED IS:

1 1. A method of providing an interface to a database, wherein the database
2 includes a plurality of classifiers of data and data linked to at least one of the
3 classifiers, the method comprising:

4 organizing a set of the plurality of classifiers into a first hierarchical data
5 structure according to a view established for a first client;
6 presenting data to the first client according the first hierarchical data structure;
7 organizing a second set of the plurality of classifiers into a second
8 hierarchical data structure according to a view established for a second
9 client; and
10 presenting data to the second client according the second hierarchical data
11 structure.

1 2. The method as recited in Claim 1, wherein the first hierarchical data
2 structure is a file system directory tree structure.

1 3. The method as recited in Claim 1, wherein the set of the plurality of
2 classifiers contains less than the plurality of classifiers.

1 4. The method as recited in Claim 1, wherein the first hierarchical data
2 structure is a file system directory tree structure according to NFS.

1 5. A method of presenting data in a database, wherein the database
2 includes classifiers(1:M) of data and data linked to at least one of classifiers(1:M),
3 wherein "M" is any positive number including 1, the method comprising:
4 creating a view for a client, wherein the view organizes a set of the classifiers
5 (1:M) into a hierarchical data structure according to characteristics of
6 the client;
7 presenting data to the client according to the hierarchical data structure.

1 6. The method as recited in Claim 5, wherein the hierarchical data
2 structure is a file system directory tree structure.

1 7. The method as recited in Claim 5, wherein the database includes N
2 classifiers and N is a number greater than M.

1 8. The method as recited in Claim 5, wherein the hierarchical data
2 structure is a file system directory tree structure according to NFS.

1 9. A method of presenting data in a database comprising:
2 receiving a request from a client;
3 translating the request into a database query;
4 producing a set of data according to the database query, the set of data
5 organized in a hierarchical data structure; and
6 presenting the set of data to the client.

1 10. The method as recited in Claim 9, wherein translating the request into
2 a database query comprises:
3 looking up the file handle in an ILocation table to obtain an ILocation; and
4 formulating a database query to query a defined set of data, the query
5 formulated from the ILocation.

1 11. The method as recited in Claim 10, wherein formulating a database
2 query to query a defined set of data comprises:
3 determining if the ILocation contains bound classifiers, each bound classifier
4 being bound by a constraining value;
5 if the ILocation contains bound classifiers, adding a clause to the database
6 query for each bound classifier in the ILocation, each clause
7 formulated to eliminate data from the defined set of data that has a
8 defined value corresponding to the constraining value of the bound
9 classifier;

10 determining if the ILocation contains unbound classifiers;
 11 if the ILocation contains unbound classifiers, adding a first clause to the
 12 database query for the first unbound classifier, the clause formulated to
 13 produce a listing of distinct values set for the defined set of data, the
 14 distinct values corresponding to the first unbound classifier, and
 15 further, adding a second clause to the database query for the first
 16 unbound classifier, the second clause formulated to select data from
 17 the set of defined data that has the value of the first unbound classifier
 18 not set to a value; and
 19 if the ILocation does not contain unbound classifiers, adding a clause to the
 20 database query that selects all data in the defined set of data.

1 12. The method as recited in Claim 10, wherein formulating a database
 2 query to query a defined set of data includes formulating an SQL query, comprising:
 3 determining if the ILocation contains bound classifiers, each bound classifier
 4 being bound by a constraining value;
 5 if the ILocation contains bound classifiers, adding a WHERE clause with a
 6 condition to the SQL query for each bound classifier, wherein the
 7 condition of each WHERE clause is set to the constraining value
 8 corresponding to each bound classifier;
 9 determining if the ILocation contains unbound classifiers;
 10 if the ILocation contains unbound classifiers, adding a SELECT DISTINCT
 11 clause to the SQL query for the first unbound classifier and further
 12 adding a WHERE clause with a condition to the SQL query for the
 13 first unbound classifier, wherein the condition of the WHERE clause is
 14 set to NULL; and
 15 if the ILocation does not contain unbound classifiers, adding a SELECT clause
 16 to the SQL query.

1 13. The method as recited in Claim 9, wherein producing a result using the
2 database query, the result formatted according to the NFS protocol, comprises:
3 querying the database with the database query that produces ILocations and
4 files;
5 translating the ILocations into unique file handles; and
6 storing the mapping from the unique file handles to the ILocations in an
7 ILocation table.

1 14. The method as recited in Claim 13, wherein translating the ILocations
2 into unique file handles comprises:
3 translating the ILocations into single byte arrays;
4 cryptographically hashing the single bye arrays into small byte arrays; and
5 padding the small byte arrays with bits to make the small byte arrays the
6 proper length of a file handle according to the NFS protocol.

1 15. The method as recited in Claim 9, further comprising providing a file
2 handle to a client upon an initial access request from the client, the file handle
3 corresponding to a view in the database, wherein the view defines an amount of data
4 in the database that is observable by the client.

1 16. The method as recited in Claim 15, wherein the view defines a
2 directory structure observed by the client.

1 17. A hierarchical display of data classification information, wherein the
2 data classification information represents classifiers of data, the data is contained in at
3 least one database, the data is at least a portion of all data contained in the at least one
4 database, and each datum is classified by at least one classifier, the hierarchical
5 display comprising:
6 a graphical display of the data classification information, wherein the
7 graphical display indicates a hierarchy of the classifiers of the data.

1 18. The hierarchical display as in claim 17 wherein the hierarchy is
2 selectable.

1 19. The hierarchical display as in claim 17 wherein the graphical display is
2 in the form of a directory tree-like display.

1 20. The hierarchical display as in claim 17 wherein the database is
2 remotely disposed from a client system, the hierarchical display further comprising:
3 a graphical display of the data classification information on a display of the
4 client system.

1 21. The hierarchical display as in claim 17 further comprising:
2 an active link associated with each classifier represented by the data
3 classification information that allows display of the data that is
4 associated with classifier.

1 22. A method of providing hierarchical data classification information,
2 wherein the data classification information represents classifiers of data, the data is
3 contained in at least one database, the data is at least a portion of all data contained in
4 the at least one database, and each datum is classified by at least one classifier, the
5 method comprising:

6 receiving a request for hierarchical data classifier information;
7 generating the hierarchical database classification information; and
8 providing the hierarchical database classification information.

1 23. The method of providing hierarchical data classification information as
2 in claim 22, wherein generating the hierarchical database classification information
3 comprises:

4 translating the request into a database query; and
5 producing a set of data according to the database query, the set of data
6 organized in a hierarchical data structure.

1 24. The method of providing hierarchical data classification information as
2 in claim 23, wherein translating the request into a database query comprises:
3 looking up a file handle in an ILocation table to obtain an ILocation; and
4 formulating a database query to query a defined set of data, the query
5 formulated from the ILocation.

1 25. The method of providing hierarchical data classification information as
2 in claim 22, wherein providing the hierarchical database classification information
3 further comprises:
4 providing the hierarchical database classification information in the form of a
5 directory tree-like structure.

1 26. The method of providing hierarchical data classification information as in
2 claim 22, wherein providing the hierarchical database classification information
3 further comprises:
4 providing the hierarchical database classification information from a server
5 system;
6 the method further comprising:
7 displaying the hierarchical database classification information with a
8 client system.

1 27. A hierarchical data structure of a database, wherein the database
2 includes classifiers(1:M) of data and data linked to at least one of the classifiers(1:M),
3 wherein "M" is any positive number including 1, the hierarchical data structure
4 generated by the method of:
5 creating a view for a client, wherein the view organizes a set of the classifiers
6 (1:M) into a hierarchical data structure according to characteristics of
7 the client;
8 organizing data into the hierarchical data structure according to the set of the
9 classifiers(1:M).

1 28. The method as recited in Claim 27, wherein the hierarchical data
2 structure is a file system directory tree structure.

1 29. The method as recited in Claim 27, wherein the database includes N
2 classifiers and N is a number greater than M.

1 30. The method as recited in Claim 27, wherein the hierarchical data
2 structure is a file system directory tree structure according to NFS.

1 31. A computer readable medium for providing an interface to a database,
2 wherein the database includes classifiers(1:M) of data and data linked to at least one
3 of classifiers(1:M), wherein "M" is any positive number including 1, the computer
4 readable medium comprising a set of instructions for enabling a computer system to:
5 organize a set of the classifiers(1:M) into a first hierarchical data structure
6 according to a view established for a first client;
7 present data to the first client according the first hierarchical data structure;
8 organize a second set of the classifiers (1:M) into a second hierarchical data
9 structure according to a view established for a second client; and
10 present data to the second client according the second hierarchical data
11 structure.

1 32. The computer readable medium as recited in Claim 31, wherein the
2 first hierarchical data structure is a file system directory tree structure.

1 33. The computer readable medium as recited in Claim 31, wherein the
2 database includes N classifiers and N is a number greater than M.

1 34. The computer readable medium as recited in Claim 31, wherein the
2 hierarchical data structure is a file system directory tree structure according to NFS.

1 35. The computer readable medium of Claim 31 wherein the computer
2 readable medium is selected from the group comprising: a hard disk drive, optical

3 drive, floppy disk drive, compact disk, or electronic signals representing the
4 instructions.

1 36. An apparatus for providing an interface to a database, wherein the
2 database includes classifiers(1:M) of data and data linked to at least one of the
3 classifiers(1:M), wherein "M" is any positive number including 1, the apparatus
4 comprising:

5 means for organizing a set of the classifiers(1:M) into a first hierarchical data
6 structure according to a view established for a first client;
7 means for presenting data to the first client according the first hierarchical data
8 structure;
9 means for organizing a second set of the classifiers (1:M) into a second
10 hierarchical data structure according to a view established for a second
11 client; and
12 means for presenting data to the second client according the second
13 hierarchical data structure.

1 37. The computer readable medium as recited in Claim 36, wherein the
2 first hierarchical data structure is a file system directory tree structure.

1 38. The computer readable medium as recited in Claim 36, wherein the
2 database includes N classifiers and N is a number greater than M.

1 39. The computer readable medium as recited in Claim 36, wherein the
2 hierarchical data structure is a file system directory tree structure according to NFS.

1 40. A system for providing an interface to a database, wherein the database
2 includes classifiers(1:M) of data and data linked to at least one of the classifiers(1:M),
3 wherein "M" is any positive number including 1, the system comprising:
4 a data processing system having a memory coupled to at least one processor,
5 wherein the memory comprises instructions for enabling the data
6 processing system to:

7 organize a set of the classifiers(1:M) into a first hierarchical data
 8 structure according to a view established for a first client;
 9 present data to the first client according the first hierarchical data
 10 structure;
 11 organize a second set of the classifiers (1:M) into a second hierarchical
 12 data structure according to a view established for a second
 13 client; and
 14 present data to the second client according the second hierarchical data
 15 structure.

1 41. The computer readable medium as recited in Claim 40, wherein the
 2 first hierarchical data structure is a file system directory tree structure.

1 42. The computer readable medium as recited in Claim 40, wherein the
 2 database includes N classifiers and N is a number greater than M.

1 43. The computer readable medium as recited in Claim 40, wherein the
 2 hierarchical data structure is a file system directory tree structure according to NFS.